Agreement error detection
in Swedish noun phrases

Master of Art Thesis In Computational Linguistics
Ola Knutsson, 1996
Supervisors: Rickard Domeij and Gunnel Källgren
Department of Linguistics, Stockholm University
IPLab, Department of Numerical Analysis and Computing Science,
Royal Institute of Technology

Abstract

This paper describes how a prototype, the agreement error detector, for agreement error detection was implemented and tested. The implementation is based on a study of Swedish noun phrases. The prototype was designed to look for agreement errors in the most frequent types of noun phrases in the study. Relaxation techniques were used for detecting the agreement errors.

A comparison was made between two different ways of handling morphological ambiguity. In the first approach words were tagged with all word classes and morphological features that they could take before being analysed by the agreement error detector. In the second approach all words were not only tagged but also probabilstically disambiguated and had only one tag before being analysed by the agreement error detector.

The results of the agreement error detection depend very much on the chosen tagging approach. With the first approach the correct detection of errors in a small error sample was about 70 percent. When using the programme on a larger unseen set of texts many false alarms were made.
The second approach shows the opposite results. The correct detection of errors in the small error sample was about 48 percent. When using the programme on the larger unseen set of texts nearly no false alarm was made.

Table of contents

0 Introduction and Overview

The main purpose of the work presented in this paper is to implement and test a computer programme for detecting agreement errors in Swedish noun phrases. This work is part of a project for developing a general error checking tool for Swedish at the Interaction and Presentation Laboratory (IPLab), at the Department of Numerical Analysis and Computing Science, Royal Institute of Technology, Stockholm, Sweden. Detection of agreement errors in noun phrases is part of the aim of the project "Computer support for Grammar Checking in Swedish" . Kerstin S. Eklundh is in charge of the project in which Rickard Domeij and Stefan Larsson also work. Prof. Gunnel Källgren at Stockholm University and Rickard Domeij are supervising this essay.

In Chapter 1 problems and objectives of the present work will be described. Chapter 2 will try to give a background to grammar checking and something about related approaches. A survey of the Swedish agreement system and a study of Swedish noun phrases are presented in Chapter 3. The implementation of the system will be discussed in Chapter 4. Chapter 5 will present the results from testing the programme. The essay will end up in Chapter 6 with conclusions and directions for future work.

1 Problems and Objectives

1.1 Detection of agreement errors in Swedish noun phrases

Ideally, an agreement error detector should find all agreement errors, and those exclusively, in Swedish noun phrases.
The agreement error detector presented in this work is limited to detect only the most frequent noun phrases and check whether these contain any agreement errors. To get the frequencies of the most common noun phrases, a study of Swedish noun phrases was made.

1.2 Robustness, overgeneration and how to deal with unrestricted text.

One goal is to design a programme that is efficient and robust enough to deal with real text. To get more robust the grammar rules are non-recursive. A complete parse is not made of a whole sentence, syntactic analysis is only made on phrase level.
A broad coverage lexicon with a morphological analyzer is needed as a base for dealing with unrestricted text. The disambiguation was solved in two different ways for comparison:

Approach A: Morphological analysed and ambiguously tagged text was used as input to the agreement error detector. Disambiguation was made with rules in the agreement error detector.

Approach B: Disambiguation was made by a probabilistic part-of-speech tagger before agreement error detection.

1.3. Error correction suggestions

In addition to detecting agreement errors, the agreement error detector should also extract morphological information to a module for generation of correction suggestions. This module is not implemented in this work but the input has to be discussed when implementing the agreement error detector, so that the module gets all information needed.

2 Background

Grammar checking is not only about error detection techniques; it is very much about computational linguistic methods in general - parsing and disambiguation techniques.
The way to a full grammar checker implementation is long if a full implementation means a system that handles the full spectrum of grammatical errors, with the necessary morphological, syntactic and semantic analysis. At this moment a total syntactic analysis is not possible. There is no powerful parser that is necessary for a grammar checker in Swedish, but there is research in this area. The Swedish version of Constraint Grammar (CG) [Karlsson et al, 1995] will soon be available and it will be very interesting to see how it is going to work on ill-formed input.

All over Europe grammar checking systems are developed for at least the official language in each country. There are the GramCheck project [Ramirez, 1995] and the French Critique [Chanod, 1993] just to mention a few.
Obviously, the grammar checker for English is most desired and indeed MSWord and WordPerfect does have one. However, there is none for Swedes using their own language. (For an overview of grammar checkers see Thollander, 1992.)

2.1 The French Critique and the relaxed approach

A relevant approach for the present work is the so called relaxed techniques and those are used with some success in the French version of the IBM Critique (a grammar checker for English) which is described in [Chanod, 1993]. The French Critique is based on principles described in [Heidorn et al, 1982]. Ungrammatical input should, as far as possible, be interpreted and errors are detected by so called relaxed techniques. This approach will not only be valid for grammar checking but also for any system robust enough to deal with unrestricted text.

The relaxed approach does not mean that everything in the language is allowed, it means that some smaller parts of ill-formedness will be accepted when there is need - to be able to get an interpretation at all. The relaxed approach and the best fitted parsing approach are supported by F. Karlsson et al [Karlsson et al, 1995]. Their second design goal of the Constraint Grammar is: "The parser should assign some structure to every input". They mean that this is a requirement for a robust approach. When using relaxed constraints in CG you also get a best fitted parsing. That is not the case when using a phrase structure grammar formalism. They also mean that the

relaxed approach should be used carefully and as a last possibility to get an input interpretation.

This is how relaxation works in the French Critique:
When the parse fails, one may assume that there is an error. Common errors in French are gender, number or person agreement.
This group of errors is handled by rules called E-conditions (E for error). All E-conditions are relaxed and this is done because there is no knowledge about the features - which are most frequently wrong. When an error is found, it is marked and the parse can go on and complete the parse tree. There are some problems, though, with the E-conditions because all are relaxed at the same time. The E-conditions will cause some unwanted analysis. One conclusion of this is that there is often more work in controlling the unwanted effects of a rule, than in writing a rule that describes a linguistic pattern correctly.
One of the most important statements of Chanod's work, which is also a defence for the relaxed approach:

> one should not implement rigid linguistic constraints that are verified only in the standard and somewhat simplified language that many small test corpora provide.
> Moreover, grammaticality should not be used as a control device during the parsing phase. The role of the parser is not to discard ungrammatical sentence, but to compute the likeliest structure of any given input text, regardless of its grammaticality. It is only at a later stage, after the syntactic structure has been computed, that grammaticality should be considered, for a possible comment.
> [Chanod, 1993, p. 106]

3 Linguistic background and studies

This section will discuss the Swedish agreement system, grammar mistakes and present a study of Swedish noun phrases.

3.1 The Swedish Agreement system

There are three different situations of agreement in Swedish. The order of these three types below is also the order of their grammatical complexity. Only the first and simplest type (number one below) will be discussed in this work.

1. Agreement in the noun phrase (phrase level). Gender, number and species have to agree in the noun phrases.

*Jag ser **den lilla hunden** (I see the* <utr.sin.def>*little* <utr/neu.sin.def > *dog* <utr.sin.def.>*)* .
*De ser **de små hundarna** (They see the* <utr/neu.plu.def .> *small* utr/neu.plu.ind/def> *dogs* <utr.plu.def>*).*

2. Agreement between the subject and the adjective in the predicative complement (sentence level):

bilen *är* ***stor*** *(the car* <*sin.*> *is big* <sin.>)
bilarna *är* ***stora*** *(the cars* <*plu.*> *are big* <plu.>).

3. Anaphoric reference (text level) The pronoun refer to an earlier word or phrase with which the pronoun has to agree.

Har du köpt ***huset****? Ja* ***det*** *är* ***mitt*** *nu.*
(*Have you bought the house* <neu.sin.def> ? *Yes it* <neu,.sin.def> *is mine* <neu.sin.def> *now.*)

Har du läst ***tidningarna*** *? Ja* ***de*** *var tråkiga.* (Have you read the papers <utr.plu.def> ? Yes they <utr/neu.plu.def > were boring)

3.2 Trying to define grammar mistakes
There are many ways to make grammar mistakes. Maybe it seems easier to define the correct noun phrases and leave the others ill-formed, but the distinction between the grammatical and the ungrammatical construction is not easily made. Sometimes it is not possible without context. Correct noun phrases could be defined as phrases where there is an agreement between the nominal features. This is not enough in Swedish, the context could make some phrases correct (1) and others not (2) for example.

1. A noun phrase before a subordinate clause:

den kvinna som jag träffade igår var trevlig, (the woman I met yesterday was nice)

2. A noun phrase after a verb:

jag träffade den kvinna igår (I met the <def.> *woman* <indef.> *yesterday) .*

To catch these context sensitive errors, a complete sentence parse probably must be done. In this work parsing is done only within the phrase. So the second example above would go undetected, the system can not find it. You have to be very careful when stating that one phrase is ill-formed, it could be correct in another context. The following two examples by Domeij[PC] proves that statement.

1. *det (it)* as a pronoun and the sentence is correct:

finns det röda bollar? (*are there any red balls?)*

2. *det (it)* as a determiner and the sentence is wrong

jag kastar det röda bollar (I throw the <neu.sin.def> *red balls* <utr.plu.indef>*)*

The conclusion is that sometimes errors made in the phrase, must be seen at sentence level. With a good disambiguation, this problem, hopefully, will not appear (see 4.1).

There is also a third situation:
3. *finns det röd boll? (are there red ball?)*

det (it) as a pronoun and a noun phrase with a missing determiner. The correct phrase could have been sentence number 1 or 4:

4. *finns det en röd boll? (is there a red ball?)*

3.2.1 The loss of gender, number or species in Swedish noun phrases
Agreement errors made in Swedish noun phrases (of type no.1 in 3.1) is the main issue of this work. Here are some examples of these errors:

en liten bilar (a <sin.> small <sin.> cars <plu.>) number error - the noun (*bilar*)

det lilla gröna stugan (the <neu.>*little green cottage* <utr.>*)* -gender error - the determiner(*det*)

en lilla bilen (a <indef.>*small* <*def.*> *car* <def.>*)* -species error - the determiner (*en*)

3.3 A study of Swedish noun phrases
One reason for conducting this study was to get an empirical material of the frequency of different types of Swedish noun phrases for the implementation. The material was also used to test the programme - to make a comparison between manual and automatic annotation.

Everything that looked like a noun phrase with agreement was documented and counted. Single nouns that, of course, are also noun phrases were not counted because the obvious lack of agreement in them. Sometimes it was difficult to say if there were two solid noun phrases or two in one. I decided to count two noun phrases in one as one simple noun phrase - to avoid recursion in the grammar rules. This kind of complex noun phrases were not very frequent. There were not many errors in the texts mainly because they were well proof-read. One disadvantage with that, was that I only got information about the frequency of correct phrases and not about the ill-formed ones.
There could be differences between them, but Domeij´s study of agreement errors in noun phrases [Domeij et al, 1996] shows that the errors very often occur in the most frequent noun phrases.

Eleven essays and 10 newspaper articles were studied. Totally 811 noun phrases were found in the texts, 284 in the essays and 527 in the newspaper articles. The texts contained about 10 500 words. There were more noun phrases in the newspaper articles and maybe the frequent use of noun phrases is significant for the professional writers who like to describe their nouns with adjectives.
A few noun phrases dominate and their frequencies in the texts are the following:

Noun phrase type	In newspapers	In essays	Example
np -> determiner adjective noun	28,8%	21,8%	*den lilla bilen (the small car)*
np -> determiner noun	21,4%	19,0%	*den bilen (that car)*
np -> adjective noun	18,8%	19,7%	*gröna ideer (green ideas)*
np -> determiner adjective adjective noun	3,2%	7,0%	*en liten röd bil (a little red car)*
np -> noun (genitive) noun	4,4%	2,8%	*pojkens boll (the boy's ball)*
np -> cardinal noun	3,2%	4,2%	*två bilar (two cars)*
np -> proper name(genitive) noun	2,7%	8,4%	*Carls uppsats (Carl's essay)*
np -> noun conjunction noun	3,5%	1,9%	*hus och bilar (houses and cars)*
Other kind of different NPs	14%	15,2%	

Table 1. Noun phrase types and frequencies

The study is too small to give more than a rough picture about the frequencies and it would be very interesting to make a larger study in a corpus such as Stockholm-

Umeå-Corpus (SUC) [Ejerhed et al, 1992] for instance. One frequency that perhaps is unusual is the 8,4%-frequency of the NP -> proper name noun - all the essays were about two newspapers and their proper names were very often used.
There were many phrases which only appeared one or two times. Implementing every single rule is not relevant, there is need for some more general techniques for handling these frequently rare constructions. Together they constitute a fairly large share of the total set of noun phrases and this is important to keep in mind when testing the programme.

One thing that could have been done was to manually tag the noun phrases in their electronic form. The automatic annotation could then have done the same thing and the results could quickly be compared. But the agreement error detector does not tag text it just extract information to send to the correction suggestions module. But with some changes the agreement error detector could annotate texts with noun phrase tags if there is need for deeper studies in the frequency of the noun phrases.

3.4 The need of an error corpus

An error corpus is a collection of texts with real errors in. There are at least three reasons for starting to build an error corpus:

I. To get the frequency of the errors and what the errors look like.

II. For knowing how to write an error grammar. An error grammar is a collection of relaxed grammar rules.

III. Most important of all - to test and evaluate the error detection programme on authentic errors so that the error detector is not designed from a small corpus and believed errors.

4 The System

The main modules of the system is here graphically presented and so are also the two different approaches. The correction suggestions are not newly generated words, but the wrong word and the correct linguistic features are presented. As a last module in the system, not implemented in this work, there should be a module for generation of the correct words. When implementing the agreement error detector the results from the study of Swedish noun phrases are used. About 80% of the noun phrases in the study are detected.

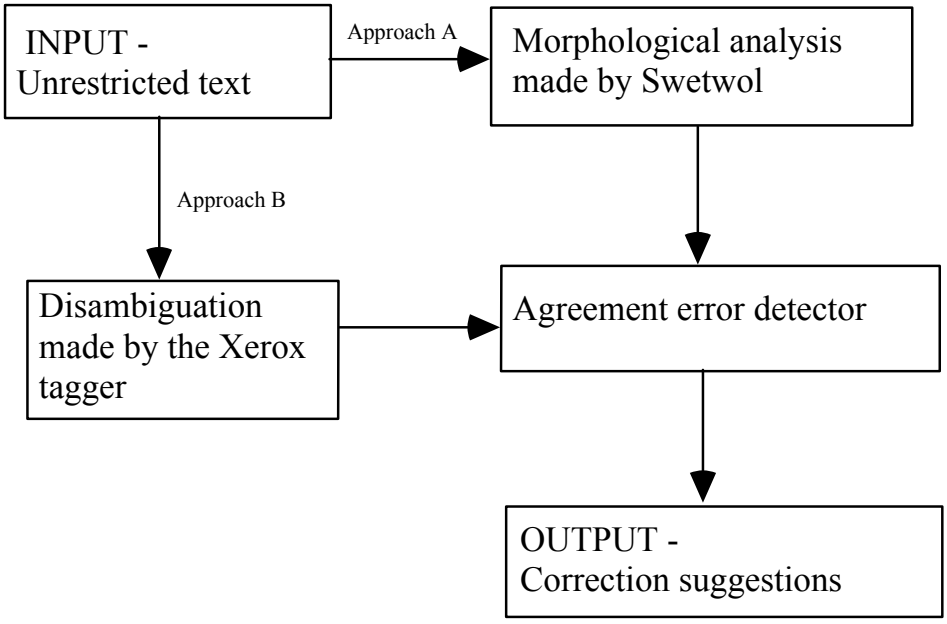

Figure 1. System overview

4.1 Two different tagging approaches

In this work one of the goals is to detect agreement errors in real-life unrestricted texts like business letters and student essays. To make the agreement error detection possible on these texts, a morphological analysis must be done. The morphological analysis should assign the words information about part-of-speech and morphological features. In this work, the morphological features should be gender, species, number and case. These morphological features should be enough as input to the agreement error detector. In this work there are two approaches - A and B. Approach A is when the system is using Swetwol ambiguous tagged text as input. Approach B is when the system is using the Xerox part-of-speech tagger (XPOST) to tag and disambiguate text. In the result section the results of using the two different approaches are presented. In the following sections the two approaches are described.

4.2 Approach A - Swetwol tagged but not disambiguated text as input

Swetwol is a system which can morphologically analyse unrestricted Swedish text, with great success [Karlsson, 1992]. Presently we are using Swetwol over Internet.

The implementation is done in the framework of Koskenniemi's two-level-model, TWOL [Koskenniemi, 1983]. TWOL has almost become a standard method for this kind of analysis. It is e.g. described in Covington's interesting book: *Natural Language*

Processing for Prolog Programmers [Covington, 1994]. Examples of output from Swetwol are shown in 4.2.1 and 4.4.5. Swetwol assigns words with all possible tags they can take in different contexts. This is both good and bad, good because there should not be any problems to tag ill-formed input and bad because that the disambiguation problem is still left and has to be solved.
Some minor problems have appeared:

- the headings in the text is not separated from the rest of the text. Maybe this is not a Swetwol related problem but it could cause some problem when parsing a sentence.
- the size of the texts which should be processed is limited.

4.2.1 Disambiguation with rules of the Swetwol tagged text

Since Swetwol tagging is ambiguous, a discussion about the disambiguation of tags has to be taken. The first disambiguation is done in the tokenizer. This programme only makes lexical entries of constituents that can be part of a noun phrase. Articles, cardinals and some pronouns (see Appendix) are treated as determiners. Present participles and past participles are treated as adjectives. A great problem though is when there are ambiguities of the same part-of-speech
Then, a noun could get two or more interpretations. Which interpretation to choose is difficult to decide, especially when the system allows ill-formed input. An erroneously picked constituent could make the parse go wrong and the result will be a false alarm signal of an agreement error in a noun phrase. With the unification in Prolog this is usually solved. Prolog first tries the "correct" interpretation of the phrase and if succeeding no correction suggestion is made. Hopefully there was no error. The phrase *samma statiska text* (*the same statical text*) is a good example of this. In this phrase the adjective generates the wrong parse if the agreement error detector would have chosen the first interpretation of the word. The Swetwol tag:

```
("<statiska>"
        ("statisk" A UTR/NEU DEF/INDEF PL NOM)
        ("statisk" A UTR/NEU DEF SG NOM))
```

The stronger disambiguation is done by the grammar rules in the agreement error detector. When this programme is trying to build an NP it may have the scope of two, three or four words. The constituents also have to agree, more or less. The difficult part will be when two constituents are relaxed at the same time. (For results on this disambiguation, see Chapter 5). To improve the results of the disambiguation rules there are two ways to go or maybe both could be used at same time:

1. Writing preventing rules like: **rule-> determiner verb <infinitive>**. This is a difficult work and you have to be very careful when choosing which constituents to include in the rules. To get rid of an over generated noun phrase like *en själv (one myself)* in the sentence: *jag såg en själv igår.* (*I saw one myself yesterday*) a rule like : **rule -> pronoun pronoun** which throws away the two words *en själv* (a self) will prevent such overgeneration. Such rules have to be well studied before using them. In this study they are only slightly investigated. The Swetwol tags of the above sentence:

```
("<jag>"
      ("jag" <VSUSSP> PRON UTR DEF SG NOM)
      ("jag" N NEU INDEF SG/PL NOM))
("<såg>"
      ("se" V ACT PAST)
      ("såg" N UTR INDEF SG NOM))
("<en>"
      ("en" ADV)
      ("en" NUM UTR INDEF SG NOM)
      ("en" N UTR INDEF SG NOM)
      ("en" <ISUSOP> PRON UTR INDEF SG NOM)
      ("en" ART UTR INDEF SG NOM))%
("<själv>"
      ("själv" PRON UTR DEF/INDEF SG NOM)
      ("själv" N NEU INDEF SG NOM))
("<igår>"
      ("igår" ADV))
```

2. The second way to disambiguate is to have a scope of three or four constituents when searching in the input string. If there is a verb before the noun phrase you may be more certain that you are actually dealing with a noun phrase. Although not yet implemented, this idea will cause no difficulties. The rule will have the following design: **rule -> verb determiner adjective noun**. The disadvantage with this, however, as an unwanted effect, is that the set of rules will grow very fast. Preventing rules are preferable because they are more general.

4.3 Approach B - disambiguation with the Xerox tagger

The second approach is to use a morphological disambiguator called the Xerox part-of-speech tagger (XPOST). The tagger assigns the word with only one tag. The Xerox part of speech tagger can be obtained from Xerox, free of charge. It has been ported to Swedish twice, first by Cutting [Cutting, 1993] and then by Eriksson and Svensson [Svensson, forthcoming]. We will use the latest version developed by G. Eriksson and T. Svensson in 1994 and 1995. I will discuss how XPOST works very briefly (for more detailed information about XPOST see [Cutting et al, 1992]).

XPOST is a probabilistic tagger which uses a Hidden Markov Model (HMM). An HMM is a set of states in a chain and every word in a string is a state. Every transition in the Markov chain is associated with a transition probability. To get these transition probabilities the model has to train on a corpus, it is also possible to help the HMM with biases about the transitions.

The biases are improving the result. Without special knowledge about the transitions you can get statistics from counting the different word tag transitions in an already tagged corpus. One problem when writing biases are that in every transition you describe there is agreement between constituents (if it is possible) and when disagreement phrases are used as input there could be problems for the model. The model will try to avoid this situations of disagreement if it can and choose another tag on the word, e.g. an adverb instead of an adjective. This model related problem could be forced by deleting some lexicon entries and only save the most frequent

form of a word. You cannot dismember the lexicon too much, however, since other constructions will get strange tags.

An example bias which describes the transition between the determiner and adjective in e.g. *den vita bilen* (*the white car*) is:
(:valid (:dt.utr.def.sin :jj.utr/neu.def.sin))
This bias says that a determiner will often be followed by an adjective.

The design of the tags is done in the lexicon. In this work, tags associated with agreement in noun phrases are of most interest. The other tags are only part-of-speech tags with no other information. The discussion of the advantages and disadvantages is not conducted here since the interest is focused on the noun phrase part of syntax.
The lexicon used in this version is extracted from SUC and contains 49830 words. The lexicon only contains the inflected forms of the words and not all words in Swedish. But a lexicon which also contains the lemma can surely be made and is necessary for generation of word suggestions.
After training the model to reach about 93% accuracy I decided to try it on ill-formed input. First on a test sample to see exactly how it reacts on special erroneous constructions (see 5.3) and then on Domeij's error sample (see 5.2).

A very important part of XPOST is the word class guesser because some words are not represented in the lexicon. The word class guesser is also trained on a corpus to learn what word suffixes are associated with what special tag. With the word class guesser the tagger could tag unrestricted texts in a more robust way. A very small test was done on one of the essays from the study and the total accuracy was 89%, but this has to be tested on a larger unknown corpus. At the moment it is not clear how much the word class guesser is improving the results, this has to be studied further.

Examples of output from the Swedish version of XPOST:

jag PN.UTR.SIN.DEF	- pronoun
ser VB	- verb
en DT.UTR.SIN.IND	- determiner
flickorna NN.UTR.PLU.DEF.NOM	- noun

4.4 Agreement error detection

In this section implementation issues of the agreement error detector will be discussed and also the implementation of the agreement error detector.

4.4.1 Parsing without a parser

One method to detect agreement errors is using phrase structure grammar rules and unification. But for Swedish there is no parser robust enough for this purpose. A best fitted parser is needed, a parser similar to the parser used in the English and French versions of the Critique. The approach of the agreement error detector is something inbetween pattern matching and phrase structure grammar formalism. The

agreement error detector is not using any parser at all if you don't count the implicit parsing mechanism of the Prolog language. The core of the agreement error detector contains a grammar with rules, and these could also be seen as patterns, but the unification in Prolog reduces the amount of patterns to implement.
One advantage with the formalism used in this work is that the rules can be used with a best fitted parser (at least the ideas and linguistic structure of the rules) and that will perhaps be a future step to improve the result of the agreement error detection. One disadvantage is that the syntactic analysis is made context free.

4.4.2 The complexity of the error possibilities

It is very easy to calculate ways to make errors in noun phrases and from a theoretical point of view there are indefinite ways to make errors. The complexity of this problem is exponential and the grammar of the different relaxed grammar rules are increasing very fast at least 2^n. This is easily proved, the features in grammar rules in Swedish are binary, the number is singulars or plurals, the species is definite or indefinite and so on. If you have a part of speech with three features which must be combined at the same time, there are, if they are binary, 2*2*2 different alternatives to be written. A very simple rule like np -> dt n, could be written in 64 different ways. This is theoretical and in real life the number of relaxed grammar rules will be less.
It is of great interest to study the rules "used" by the user and the errors made. It is no use to write a lot of relaxed grammar rules which will never be used. Such rules will slow down the system and most important they will detect noun phrases that are not to be seen as erroneous noun phrases.

4.4.3 Rule generalisation

For a linguist it is interesting to know what kind of errors writers make. But the end user is not interested in such things. The end user would probably not improve their texts by getting exactly the morphological feature that is wrong. So for the users sake the grammar rules could be more generalised. The grammar rules could be generalised to such a degree that the morphological generator gets all the information it needs. Obviously the generator needs only the correct features of the generated word. In other words – the wrong features are not of interest for the generator.

But rules too relaxed may cause some overgeneration, therefore it is good to have knowledge about the errors that can be made. A study made by Domeij [Domeij et al, 1996] points out that the most common errors, are those with only one feature that does not agree e.g. *en liten bilar* (a <sin.> small <sin.> cars <plu.>). This knowledge could perhaps be used when writing relaxed rules with only one feature relaxed. However, for knowledge about the possibility of stopping overgeneration, testing of unknown texts must be made.

4.4.4 The correction of agreement errors

It is very hard to know what a writer intended when making an error, sometimes it is impossible. The correction suggestions in a grammar checker system is a very important part and many users cannot make any sense of the output from the grammar checker if it is presented as linguistic features. There are two things to consider here: The first thing is how to get the right correction suggestions, which could not be answered really without further studies, guesses have to be made. The second thing is to design the grammar so that the most frequent constructions are detected first and so that longest constructions are analysed first (see 4.4.5.2). The programme will, for the sake of robustness, only make one analysis and sometimes the correction suggestions may be wrong. There may be other cases when the programme makes a stupid suggestion – maybe there is need for semantic information to disambiguate the phrase. The rules have to be tested a lot and there will always be exceptions from the normal detection formula (see the result section for further discussions).

When designing the grammar rules the most practical thing is to have the noun as the head of the phrase and the other constituents as modifiers. It means that when the gender feature of the noun does not agree with the other constituents, the noun should be left unchanged. In e.g. *den lilla huset* (the <utr.> little house <neu.>) the noun could not be changed to agree with the other constituents, the whole word must be changed which is a very complicated task. Change the other constituents instead, take the lemmas of *den (that)* and *lilla (little)* and the features of the noun and then the correct words can be generated.[1] From this follows that the noun should be changed when there is no agreement in number or species. In e.g. *en stor villan* the wrong part of the noun phrase is the species of the noun. In e.g. *de stora villa* the number of the noun should be changed (and maybe also the species feature). All this comes from the assumption that if there is agreement between two constituents, e.g. the determiner and the adjective, the errors belong to the noun. If there is agreement between the adjective and the noun, the determiner has to be changed. The hierarchy of these rules could be discussed but until a large study of error frequencies is made, the rules with erroneous determiner and/or adjective will be placed before the noun's.

4.4.5 The internal error detector system

The two modules of the internal error detector system will be discussed in this section. There are two versions of every module, one for Swetwol input and one for XPOST input. Only the Swetwol version will be discussed since it was first documented. The XPOST version is much like the Swetwol version but a little bit more simple.

[1]There is no normal lemma for *den* , because *den* could not be replaced without changing the gender (in this case) and that means a new word. A special module for the determiners would probably be needed.

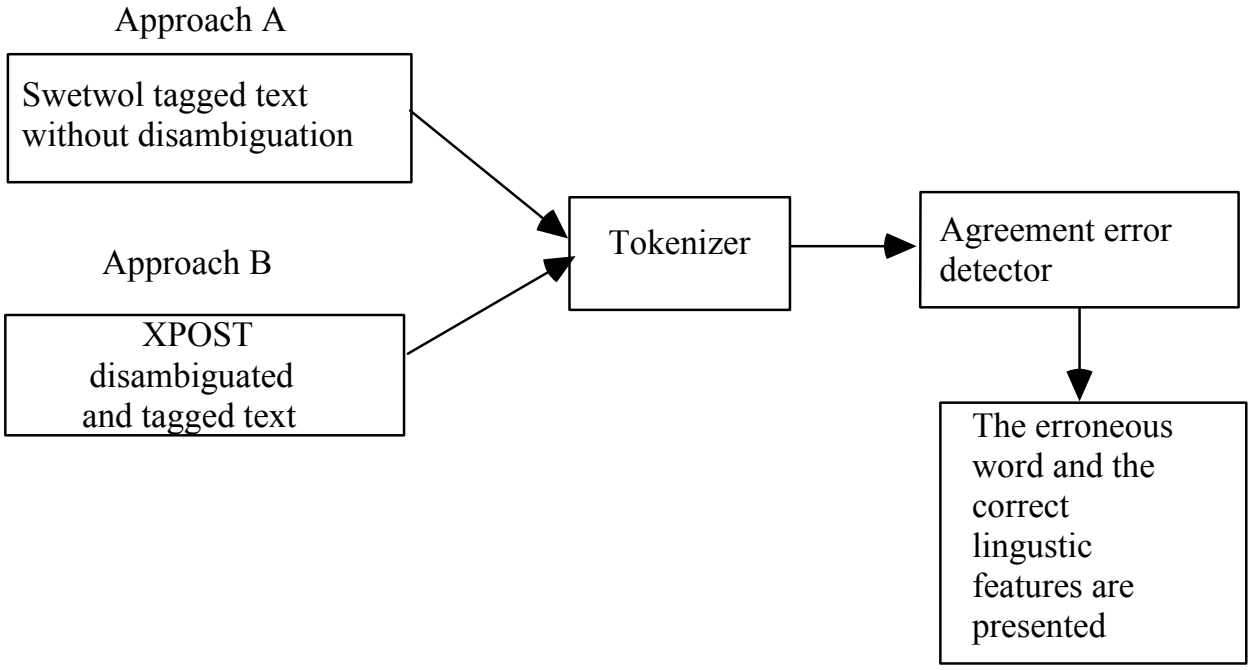

Fig. 2 The internal error detector system

4.4.5.1 The tokenizer

The tokenizer should process Swetwol tagged text in two ways: first the words and the morphological information should be asserted as lexicon entries to be used in the parsing process and as a second process the tokenizer should extract the sentence to process in a list. After these two processes each sentence can be parsed. Each word is tagged on a one-word level and the tag contains all ambiguities for the word. The word *för* gets the following analysis:

(“<för>“
(“för“ PREP)
(“för“ AD-A)
(“för“ ADV)
(“för“ N UTR INDEF SG NOM)
(“för“ KONJ)
(“föra“ V ACT PRES)
(“föra“ V ACT IMP))

The Swetwol analysis is in a LISP-notation and has to be transduced to Prolog. Ideas from another tokenizer made by Lindberg and Santamarta [Lindberg and Santamarta, 1994] influenced the implementation of the tokenizer in this work. Their

tokenizer is dealing with a similar input, and the main idea they had, was to get to the Prolog-world as quickly as possible. This is also done in this work.
The output in Prolog code for the word *för:*

term('för', [för, prep], [för, ada], [för, adv], [för, n, utr, indef, sg, nom],
[för, konj], [föra, v, act, pres], [föra, v, act, imp]).

There is a module in the tokenizer witch is called *between* and this function takes care of the interaction between the tokenizer and the agreement error detector. This is done by reading the output file from the tokenizer. Every Prolog term is read and the first word is put on a list. Only words that have anything to do with noun phrases are asserted in a temporary lexicon. Every term is looked up and if one contains an analysis which could possibly be a constituent of a noun phrase this analysis is asserted. After every sentence is parsed, the lexicon terms are retracted.

All the articles and some of the pronouns will be treated as determiners. Some pronouns can never be used as determiners. The present participle and the perfect participle will be treated as adjectives. Some of these treatments will probably cause overgeneration of erroneous noun phrases, especially when the input is Swetwol tagged text and for example there is no disambiguation of participles and verbs.

4.4.5.2 The agreement error detector

The agreement error detector detects the most frequent noun phrases and tries to find agreement errors. Without the relaxed rules the agreement error detector could be used as an NP-detector.
There are existent noun phrase detectors for example Voutalainen's NPtool [Voutalainen, 1993]. Voutalainen means that there are several uses for an NP-detector: as a pre-processor for more advanced parsing, automatic large-scaled analysis of running text and for translation units.

The grammar rules are not recursive since it is very seldom necessary and also time consuming for this type of analysis. The recursive rules also make the system much more complex than wanted.
The programme checks if a string of words could be a noun phrase. The first word in a string is tested if it is able to begin as a noun phrase. If and only if it can, the next word is tested to see if it could be the second constituent of the NP and so on until one noun phrase is completed. First there are rules for correct input, if they all fail, the programme will try the relaxed ones to find the errors. In other words, the search for noun phrases and agreement errors are done in the same process.

With the Swetwol tagged input the hierarchy of the rules is very important, because then the disambiguation process will be done in the agreement error detector (see also 4.4.4). A rule like **NP -> determiner adjective adjective noun**
has to be checked before a rule with only one adjective. There can be ambiguities between adjectives and nouns. The phrase *den vemodiga finska mentaliteten (the*

melancholy Finnish /Finn mentality) is a correct noun phrase and should not get an analysis of the sometimes erroneous noun phrase with only one adjective – *den vemodiga finska*. (*the* <def.> *melancholy Finnish* <indef.>).

The Swetwol tags look like:

```
% The determiner
("<den>"
    ("den" <2*US*> PRON UTR DEF/INDEF SG NOM)
    ("den" <2*NSG*> ART UTR DEF SG NOM))

% The first adjective
("<vemodiga>"
    ("vemodig" A UTR/NEU DEF/INDEF PL NOM)
    ("vemodig" A UTR/NEU DEF SG NOM))

% The second adjective or the erroneous noun
("<finska>"
    ("finska" N UTR INDEF SG NOM) % noun
    ("finsk" A UTR/NEU DEF/INDEF PL NOM) % adjective
    ("finsk" A UTR/NEU DEF SG NOM)) % adjective

% The noun
("<mentaliteten>"
    ("mentalitet" N UTR DEF SG NOM))
```

The hierarchy of the rules is also important when the programme is to decide what constituent that is wrong. This has to be tried out on a larger corpus. In the tentative version of this work, the noun´s features are relaxed last and this because of the assumption that the noun is most often correct in the phrase.
After being analysed, the noun phrase string is cut off from the whole sentence, and the programme starts with the next word in the sentence. If the first word is not accepted as a first constituent of a noun phrase by any rule, it will be discarded. A simple relaxed grammar rule will detect the following constructions:

en lilla flickan	- species error
det lilla flickan	- gender error
de lilla flickan	- number error
ett lilla flickan	-species and gender error
några lilla flickan	- species and number error

There is only need for one single rule for these kinds of errors. Comparison made with results from the use of one rule for each construction, have rendered a satisfying result. There are no differences in performance found. This is positive, since the number of rules will be much smaller.
As an example of how the rules look, this is the relaxed grammar rule for erroneous determiners:

```
/*
```

```
*
*  np_out
*
*  This np_out rule describes the pattern of a noun phrase with an erroneous
*  determiner. The determiner could be wrong in one, two or three features.
* The relaxed constituent, doesn't agree with the other constituents.
* The "_"- sign means a relaxed feature. The adjective and the noun have
* to agree, this is checked with unification.
* The feature's values of noun/adjective are extracted and sent as output
*
*/

np_out([X, Y, Z|Xs]):-
                dt(X, Lemma1,  [gen=_, spec=_, num=_]),
                a(Y, Lemma2,  [gen=Gen, spec=Spec, num=Num]),
                 n(Z, Lemma3, [gen=Gen,spec=Spec, num=Num, case=Case]),

                write('nominalfrasen: "', write(X), write(' '), write(Y),
                write(' '), write(Z),
                write(' " - är inkorrekt, determineraren ska bytas ut'), nl,
                write('determinerarens lemma och särdrag ska vara:
                lemma='), write(Lemma1), nl,
                write('gen='), write(Gen), nl,
                write('spec='), write(Spec), nl,
                write('num='), write(Num), nl,

                np_out(Xs).                % Xs is the rest of the sentence
```

There is also a trash rule at the end of the programme, which takes care of the constituents that can not be a part of an NP. Example on this can be *ser (see)* in *ser en liten bilen (see a little car)*. The word *ser* is cut off from the input string:

```
np_out([X|Xs]):-
   np_out(Xs).
```

Examples of a tentative output with an input from XPOST:[2]

```
nominalfrasen: "det färdiga resultat" - är inkorrekt, substantivet ska bytas ut
lemma=resultat
gen=neu
spec=def
num=sin
case=nom
```

[2] The Swetwol version would look the same with the only difference that the Swetwol version presents a real lemma. The Xerox version will present the inflected word form until there are lemmas in the Xerox lexicon

nominalfrasen: "varje psykologiskt term" - är inkorrekt, adjektivet ska bytas ut
lemma=psykologiskt
gen=utr
spec=ind
num=sin

nominalfrasen: "de senaste året" - är inkorrekt, determineraren ska bytas ut
lemma=de
gen=neu
spec=def
num=sin

5. Results

The results presented here are from the different evaluations made for testing and comparing the two approaches. It also tests how well the agreement error detector works with each approach. The agreement error detector can not be tested without the input approaches - the results are a combination of the agreement error detector accuracy and the chosen input approach accuracy.

5.1 Comparison between manual and automatic annotation

The construction of rules in the agreement error detector is based on the study of noun phrases (see 3.3). Only the most frequent noun phrases were implemented and tested for coverage.
Three of these texts (two essays and one newspaper article, 113 NPs) have been automatically annotated with the agreement error detector. Every test run was made with a Swetwol based input and also with a XPOST based input and the results are the following:

Text name	**Words**	**Manually annotated NPs**	**Automatically annotated NPs with Swetwol**	**Automatically annotated NPs with XPOST**
"tidning1"	413	31	35	20
"tidning3"	480	39	47	28
"Unionen"	415	43	43	33

Table 2. Comparison between manual and automatic annotation

Text name	**Words**	**Over generated NPs with Swetwol**	**Over generated NPs with XPOST**	**Correct annotated NPs with Swetwol**	**Correct annotated NPs with XPOST**
"tidning1"	413	10	1	25	19
"tidning3"	480	11	1	36	27

"Unionen"	415	5	1	38	32

Table 3. Continuing table 2

This is only a small test but with only a few noun phrases implemented the coverage is good. When using a Swetwol based input, the agreement error detector detects a lot of phrases but many of them are not noun phrases. The XPOST tagged input is better when running the agreement error detector, in the sense that the overgeneration of noun phrases could be neglected.

5.2 The error detection test

The text to be analysed was Domeij´s error sample. A more detailed comparison is presented in Appendix. There were 46 errors to detect.

Input base	**Detected errors**	**Correct detected errors**	**False alarms**
Swetwol	37	33	9
XPOST	27	22	8

Table 4. The detection test

The results are a bit surprising, the test sample is small and the results have to be seen as tendencies. They could be explained by the fact that
some words in the error corpus were not represented in the XPOST lexicon and consequently the XPOST has to guess. The accuracy of the XPOST was 93 percent when it tagged the error corpus. The accuracy can be improved, for this study, however, there was not enough time.
To stop the overgeneration some changes can be made: some pronouns are asserted as determiners and this cause some mistakes. There are also some noun phrases which would not be found, because the tagger chose a pronoun instead of a determiner in disagreement situations. This has to be studied further, because one of the goals of bringing a deterministic tagger into this project was to stop the overgeneration. As seen below the results are not better than Swetwol´s in this test. It must be said, that the Domeij error sample only contains sentences with errors and that is of course very abnormal, other tests, like the test in 5.3, have to be done.

The improved results from the Swetwol version can be explained by the fact that Swetwol has got a very broad lexicon and a robust morphological analyzer with high accuracy. The disambiguation of the Swetwol tags by the agreement error detector is better than first assumed. How to stop the overgeneration of the Swetwol input has already been discussed in 4.2.1 and is further discussed in 5.3.1.

5.3 The overgeneration test

The essays from the noun phrase study (5454 words) were used for testing how much the agreement error detector would overgenerate. After improving the relaxed

rules to find almost every agreement error in the most frequent noun phrases I realised other problems such as overgeneration. Overgeneration occurs when non noun phrases are detected and wrong correction suggestions are presented. The policy of the detection project is rather to try to free ill-formedness than judge it. Absolute accuracy about an error is impossible, but well-known cases where ill-formed phrases could be seen as well-formed, (as in section 3.2) should not be detected. To avoid this, the overgeneration must be stopped. This can be done with fewer relaxed grammar rules, or with a stronger and better disambiguation. Therefore a test of how much the agreement error detector will overgenerate, depending on input (XPOST or Swetwol), was made.
In this test there were 63 false alarms when using Swetwol tagged text as input and only 7 false alarms when using XPOST tagged text as input.
XPOST's overgeneration is nearly neglectful but Swetwol's has to be improved and if any improvement should be done at a later date some preventing rules which will disambiguate the input, must be added. (see section 4.2.1). You have to know in what situations there is need for preventing rules and some ideas are presented in 5.3.1.

5.3.1 Overview of the disambiguation errors with Swetwol tagged text as input
In the overgenerated noun phrases, one or more words are disambiguated wrong, but only one disambiguation error is counted – the error which probably has caused the overgeneration. Of special interest are the detected noun phrases, which are pointed out as ill-formed by the agreement error detector in a correct way. With the context, however, they should have been seen as correct ones. An example of this is the phrase *något yngre student* in *Ledaren försöker vara mer kvick och fånga intresset hos en något yngre student*. It is detected as ill-formed, and the programme have not proceeded wrong, the phrase is in my opinion correct - the word *något* should be seen as an adverb. But there is a disagreement situation: *något* is neuter and *student* is non-neuter. Special rules for special phenomena are necessary in a real application. Swetwol tagged input causes a lot of overgeneration of noun phrases and how this is possible is explained below and illustrated with examples. The amount of false alarms is also presented in the overview.

There were 10 adverbs interpreted as adjectives in the test.
Ex. *Gadden har **en betydligt större upplaga** (ca 30000) än Osqledaren (ca 9000) eftersom Universitetet har betydligt fler studenter*

Some constituents were interpreted as a noun instead of an adjective at 10 times.
Ex. *Gadden ger **ett betydligt snyggare** och proffsigare intryck när man slår upp den.*
The Swetwol tags for snyggare are:

```
"<snyggare>"
      "snygg"  <Cmp> A UTR/NEU DEF/INDEF SG/PL NOM
      "snyggare"  VDER-are N UTR INDEF SG/PL NOM
```

Noun instead of a pronoun: 8

Ex. ***Det första man*** *tänker på vad det gäller utseende är nog de olika formaten och papperskvalitén.*

Grammatical correct phrases but pointed out as ungrammatical: 8
Ex. *Troligen beroende på att den produceras av* ***enbart teknologer.***

Noun instead of a verb: 7
Ex. *Och* ***det visar*** *sig faktiskt att utsidan vittnar ganska väl om skillnaderna och likheterna, Gaudeamus mer som en slags nyhetstidning för universitetsstudenterna och Osqledaren en mer nöjesinriktad tidning.*
The Swetwol tags for visar are:

"<visar>"
"visa" V ACT PRES
"vise" N UTR INDEF PL NOM %*visar* is in swedish another word for queen bee.

Noun instead of an adverb: 5
Ex. *Detta bekräftas av att Gaddens redaktion består av ungefär* ***dubbelt så*** *många medarbetare som Osqledaren.*

Noun instead of a preposition: 5
Ex.*Tittar man på innehållet ser man att Gaudemus innehåller betydligt fler annonser, ja till och med* ***sådana för*** *071-nummer.*

Noun instead of a determiner: 3
Ex. *Gaudemus har i* ***det hela en*** *mer "seriös"(och kanske lite "torr") atmosfär.*

Adjective instead of a preposition: 3
Ex. *Som student* ***vid universitetet*** *är det högst ett par artiklar som är av intresse, eftersom Gadden täcker ett sådant brett område.*

Noun instead of a subjunction: 2

Ex. *Men trots **det så** hittar man de intressantaste artiklarna i gadden, som aldrig är tråkig till skillnad från Osqledaren som bitvis kan vara lite trist att bläddra i, men i gengäld så har den en ganska avslappnad stil i sina artiklar med en hel del slang och dylikt.*

Determiner instead of a pronoun: 2
Ex. *Bortsett från att ett lättsammare ämne ofta medför ett lättsammare språk, är **det mindre skillnad** mellan tidningarna vad gäller språket.*

5.3 Testing XPOST on ill-formed input

The main problem when using Swetwol as a base in the system, is that the agreement error detector finds more noun phrases than it should. The XPOST shows the opposite results; some agreement errors are not detected.
XPOST is tested on a test sample of constructed agreement errors in noun phrases, to see if there are any patterns of errors in the disambiguation of ill-formed input. Seventy-nine erroneous noun phrases and 16 correct phrases (for checking) were tested. Fifty-three phrases were detected as errors in a correct way and 28 errors were not detected, there were no false alarms. The most common error made by the XPOST was to tag determiners as pronouns and this is made to avoid transitions where there is a non-agreement word transition (see also 4.3). A few adjectives were tagged as adverbs by the same reasons as above. The pronoun-determiner problem can easily be solved, but then you get opposite problems – overgeneration, like in the Swetwol case. The main conflict in this work is to find the largest number of errors and have overgeneration, or to find fewer errors and have very little overgeneration. A second test was made with only the part-of-speech tags in the tagset and the results were perfect - all determiners were tagged as determiners, and adjectives as adjectives. Obviously there is a way to tag ill-formed input with very good results. The morphological information could then be found in a second lexicon. This method may seem a bit complex, but it is worth trying, however, because this is a successful way to disambiguate ill-formedness.

6. Conclusions and future work

6.1 Conclusions

The idea of using non-recursive relaxed grammar rules has in some ways been successful. They can detect agreement errors in noun phrases and make an output for generation of correction proposals. Overgeneration is the main problem. It makes very strange correction proposals and has to be stopped in a real application. False alarms are very irritating for the user. Both Swetwol and XPOST based input in the agreement error detector make proposals which are made correctly. The detected phrase is correct in the context, however, so detection could never be absolutely safe. This means that the undergeneration of detection is preferable.
Using only the most frequent noun phrases has been successful – many of the agreement errors in Domeij's error sample were found.
The results of the two input approaches are compared and for Swetwol this can be stated(+ stands for advantage and - stands for disadvantage):

+ A very broad coverage of words and new word combinations
+ No disambiguation is made and thus, special disambiguation rules for ill-formed input can be written.
+ The detection of agreement-errors is very good.

- Very uncommon interpretations of some words are represented and will interfere with the disambiguation process.
- The overgeneration of non erroneous noun phrases is very high and must be stopped.

For the XPOST this can be stated:
+ The overgeneration of non erroneous noun phrases is negligent
+ The texts are disambiguated and this will speed up processing following.
+ The system is very open and changes can easily be done

- The disambiguation process is designed for texts without any errors since errors in proof-read texts are still uncommon. The tagger wants to tag the words correctly, and thus some ill-formedness will be forgotten.
- The coverage of the lexicon is not large enough and the lexicon only contains the inflected word forms.
- The statistical model of the language that is generated is very difficult to understand.

As seen above there are advantages and disadvantages with both types of input, and if there is interest only in finding errors the Swetwol version could be preferable. I think, however, that in a real system, the overgeneration must be stopped and XOST is best for that. The Swetwol version could be improved by preventing rules, this could render very interesting results regarding robustness in the two systems. If a new lexicon for XPOST is built, with lemmas and possibilities of generation, that version will be closer to a real application. The output from XPOST is also useful for other modules in the general error detection programme.

When dealing with unrestricted text, and unknown words, XPOST is using a word class guesser. This word class guesser is based on statistical occurrences of word suffixes and only one occurrence of a suffix is enough to get a representation. This is not very good since, wrong guesses are made and then there will be overgeneration of erroneous noun phrases. One way out, is to write a new word class guesser with only a few but statistically safer, suffixes, so that only some guesses are done and the other words are left unanalysed.

There is, however, a need for other systems where the relaxed approach is necessary. When dealing with real-life input you have to take care of some ill-formed input. The problem will be enormous when dealing with speech-to-text, the necessity of pragmatism is more important for this kind of applications, than theoretical grammar models. This is also the case for grammar checking and in general, future grammars must be able to handle ill-formedness if they are to be seen as robust.

There have been some situations in this work when correct phrases are detected as errors because of the context. These situations can be avoided with a more powerful parser. If the Constraint Grammar can be useful for these situations will be of future interest. There have also been situations, where the uses of unification of features have failed.

A phrase as *något yngre student* is detected as ill-formed and here the programme has acted correctly, the phrase is in my opinion correct, but there is a non-agreement situation: *något* is neuter and *student* is non-neuter. Special rules for special phenomena will be necessary in a real application.

6.2 Future work

A lot of things need improving. Future work, however, should be concentrated on refining and completing the morphological disambiguation (see 6.2.1) together with the agreement error detection (see 6.2.2). Reasons being, that both areas are very complex.

6.2.1 Improvements of the morphological disambiguation

One conclusion in this work is that there are problems with morphological disambiguation of ill-formed input and unrestricted text. Disambiguation of ill-formed text may be solved with the following method.
Use the part-of-speech tagset for disambiguation and a second lexicon to get the morphological information from. This method will work, because there will never be any disagreement situations for the tagger to avoid.

The word class guesser in XPOST should be refined by more training, or by writing a word class-guesser, based of well-known linguistic knowledge about word suffixes. The same lexicon used by XPOST, can also be used as a module for generation of correction proposals. A lot of work can be done to improve the results. Refining the lexicon is important, since the lexicon is the base of the system. All information wanted later, comes from the lexicon. The tagset is defined in the lexicon and what tagset to use is relevant. The result a tagset will cause is very difficult to predict without implementing and testing it. Just a few new tags could change the results of the new model a lot.

6.2.2 The work with the agreement error detector is not finished

The agreement error detector must further be tested and completed with rules for special phenomena. User studies are also relevant to get ideas for improvements and for how to design the correction suggestions. Larger studies of texts are also important to get more statistically correct information about the noun phrases. Ways to make errors can also differ between Swedish native writers and writers with Swedish as a foreign language. This work has only taken Swedish native writers into consideration and the programme may be of little use for a writer with Swedish as a foreign language. The way to handle the correction suggestions, can also differ

between these two groups and suggestions will probably have to be designed for different user groups.

References

[Chanod, 1993] Chanod, Jean-Pierre, 1993. *A Broad-Coverage French Grammar Checker: Some Underlying Principles*. Proceedings of the Sixth International Conference on Symbolic and Logical Computing. Dakota State University Madison, South Dakota 57042-1799.

[Covington, 1994] Covington, Michael A, 1994. *Natural Language Processing for Prolog Programmers,* Prentice Hall

[Cutting, 1992] Cutting, Douglass R. 1992. *Porting a Stochastic Part of Speech Tagger to Swedish*. Xerox Palo Alto Research Center (Parc) and Swedish Institute of Computer Science(SICS).

[Cutting et al, 1992] Cutting, D. , J. Kupiec, J. Pedersen and P. Sibun, 1992. *A practical part-of-speech tagger*. Proceedings of the Third Conference on Applied Natural Language, Trento, Italy, April 1992. ACL. Also available as Xerox PARC technical report SSL-92-01

[Domeij et al, 1996] Domeij, R., Knutsson, O. and Larsson, S. 1996. *Datorstöd för språklig granskning under skrivprocessen – en lägesrapport*, IPLab, Department of Numerical analysis and Computing science, Royal Institute of Technology, Stockholm, Sweden.

[Ejerhed et al, 1992] Ejerhed, Eva, Gunnel Källgren, Ola Wennstedt and Magnus Åström,1992. *The lingustic annotation system of the Stockholm-Umeå Corpus project,* version 4.31.

[Heidorn et al, 1982] Heidorn G.E., K. Jensen, L.A. Miller, R.J. Byrd et M.S. Chodorow. 1982. *The EPISTLE Text-Critiquing System,* IMB system journal, vol.21, no.3.

[Karlsson, 1992] Karlsson, Fred, 1992. *SWETWOL: A comprehensive morpological analyzer for Swedish*, Nordic Journal of Linguistics, 15, 1-45.

[Karlsson et al, 1995], Karlsson, Fred, Atro Voutalainen, Juha Heikkilä, Arto Anttila, 1995. *Constraint grammar - A language -Independent System for Parsing Unrestricted Text*, Natural Language Processing 4, Mouton de Gruyter,Berlin ,New York.

[Koskenniemi, 1983] Koskeniemi, K. 1983. *Two-level Morphology: A General Computional Model for Wordform Recognition and Production*. Publications of the Department of General Lingustics, University of Helsinki, No.11.

[Lindberg and Santamarta, 1994] Lindberg, Nikolaj and Lena Santamarta, 1994. *When SUC met CLE*, Bachelor of Art Thesis, Department of Lingustics, Stockholm University.

[Ramirez 1995] Ramirez,F. 1995 *The MLAP GramCheck Project: A Bilingual Grammar and Style Checker*. A summary of Deliverable 3 of the GramCheck project: Lingustic design and implementation techniques for grammar correction", written by F.Ramirez and F. Sanchez-Le´on.

[Thollander, 1992] Thollander, T. 1992. *Datorstöd för språklig textgranskning*. Kungliga Tekniska Högskolan, Inst. för numerisk analys och datalogi, rapport IPLab 56 (TRITA -NA-P9218).

[Vosse, 1992] Vosse, T, 1992. *Detecting and Correcting Morpo-syntactic Errors in RealTexts*, ACL:111-118.

[Voutalainen, 1993] Voutalainen, Atro, 1993. *NPtool, a detector of English noun phrases*. Proceedings of Workshop on Very Large Corpora. Ohio State University.

Appendix: More details from the error detection test

Results from a test run are presented below with the generalised relaxed grammar rules on an ongoing collection of errors collected by Domeij [Domeij, 1995]. The output from the Swetwol version and the XPOST version are presented together. Over generated noun phrases are also presented. The noun phrase that is wanted is underlined. How to read the output:

% The noun phrase that has been detected is presented with a comment about the erroneous constituent (s):
nominalfrasen: "bäst importfunktionerna" - är inkorrekt, adjektivet ska bytas ut

% The lemma of the inflected word which should be used as a base for generation % of new words (no real lemma in the XPOST version):
lemma=god

% the gender feature of the word to be generated
gen=utr

% the species feature of the word to be generated
spec=def

% the number feature of the word to be generated
num=pl

% The rule in the agreement error detector which has been used, interesting in two ways: for control of the system and when comparing the XPOST version with the Swetwol version - different rules could have been used.
regel 25

1. De vanliga kriterier för att sortera brev i brevlådan är avsändare, ämne, nummer och datum.
Error analysis: This error is context sensitive and the phrase should have been correct in another context.

2. Programmen har blivit allt bättre på att läsa in texter och bilder, bäst importfunktionerna har Pagemaker och Corel Ventura. **DETECTED!**

Swetwol:
nominalfrasen: "bäst importfunktionerna" - är inkorrekt, adjektivet ska bytas ut
lemma=god
gen=utr
spec=def
num=pl
regel 25

3. EFI Color ska garantera att den färg som visas på skärmen och på korrekturutskrifterna motsvarar det färdiga resultat. **DETECTED!**

Swetwol:
nominalfrasen: "det färdiga resultat" - är inkorrekt, determineraren ska bytas ut
lemma=det
gen=neu
spec=indef
num=pl
regel 19

XPOST:
nominalfrasen: "det färdiga resultat" - är inkorrekt, substantivet ska bytas ut
lemma=resultat
gen=neu
spec=def
num=sin
case=nom
regel 21

4. Varje psykologiskt term som "tänka", "planera", "fantisera", "älska", "njuta" osv. beskriver, enligt deras uppfattning, beteendet och inget annat. **DETECTED!**

Swetwol:
nominalfrasen: "varje psykologiskt term" - är inkorrekt, adjektivet ska bytas ut
lemma=psykologisk
gen=utr
spec=indef
num=sg
regel 20

XPOST:
nominalfrasen: "varje psykologiskt term" - är inkorrekt, adjektivet ska bytas ut

lemma=psykologiskt
gen=utr
spec=ind
num=sin
regel 20

5. En ny variant av materialism, kallad dialektisk materialismen, skapades av marxistiska filosofer. **DETECTED!**

Swetwol:
nominalfrasen: "dialektisk materialismen" - är inkorrekt, adjektivet ska bytas ut
lemma=dialektisk
gen=utr
spec=def
num=sg
regel 25

XPOST:
nominalfrasen: "dialektisk materialismen" - är inkorrekt, adjektivet ska bytas ut
lemma=dialektisk
gen=utr
spec=def
num=sin
regel 25

6. Han är inte säker på att ökad statligt hjälp till företagen är rätta vägen att gå, även om företaget haft bra hjälp av exempelvis Exportrådet i vissa länder.**DETECTED!** (except *ökad*)

Swetwol:
nominalfrasen: "statligt hjälp" - är inkorrekt, adjektivet ska bytas ut
lemma=statlig
gen=utr
spec=indef
num=sg
regel 25

7. Hetast på Internet har under de senaste året varit World Wide Web, WWW.**DETECTED!**

Swetwol:
nominalfrasen: "de senaste året" - är inkorrekt, determineraren ska bytas ut
lemma=de
gen=neu
spec=def
num=sg
regel 19

XPOST:
nominalfrasen: "de senaste året" - är inkorrekt, determineraren ska bytas ut
lemma=de

gen=neu
spec=def
num=sin
regel 19

8. Användningen av sådan tjänster förväntas öka kraftigt när själva informationsinhämtandet underlättas. DETECTED**!**

Swetwol:
nominalfrasen: "sådan tjänster" - är inkorrekt, substantivet ska bytas ut
lemma=tjänst
gen=utr
spec=_90735
num=sg
case=nom
regel 23

9. Utveckling av metoder för automatisk igenkänning av översättningsekvivalenter i korpusar bestående av parallelltexter, dvs. en grundtext med översättningar till andra språk är ett särskilt intressant del av den textbaserade lexikonbyggnaden. **DETECTED!**

Swetwol:
nominalfrasen: "ett särskilt intressant del" - är inkorrekt, determineraren och första adjektivet ska bytas ut
lemma=ett
gen=utr
spec=indef
num=sg

lemma=särskild
gen=utr
spec=indef
num=sg
regel 15

XPOST:
nominalfrasen: "ett särskilt intressant del" - är inkorrekt, determineraren och första adjektivet ska bytas ut
lemma=ett
gen=utr
spec=ind
num=sin

Overgeneration by XPOST:
nominalfrasen: "den textbaserade lexikonbyggnaden" - är inkorrekt 106 determineraren och adjektivet ska bytas ut
lemma=den
gen=neu
spec=ind
num=plu
lemma=textbaserade
gen=neu
spec=ind
num=plu
regel 22

10. Ett sätt att skapa ett sådan robust beteende är att utnyttja heuristiska strategier. **DETECTED!** (except *ett*)

Swetwol:
nominalfrasen: "sådan robust beteende" - är inkorrekt, determineraren ska bytas ut
lemma=sådan
gen=neu
spec=indef
num=sg
regel 19

XPOST:
nominalfrasen: "ett sådan robust" - är inkorrekt, determineraren ska bytas ut
lemma=ett
gen=utr
spec=ind
num=sin
regel 19

11. Studien initierades av NUTEK, och dess resultat skulle bidra till att ge underlag för den framtida utveckling av språkteknologi i Sverige. **DETECTED!**
Swetwol:
nominalfrasen: "den framtida utveckling" - är inkorrekt, determineraren ska bytas ut
lemma=den
gen=utr
spec=indef
num=sg
regel 19

XPOST:
nominalfrasen: "den framtida utveckling" - är inkorrekt, determineraren ska bytas ut
lemma=den
gen=utr
spec=ind
num=sin
regel 19

12, 13. Vilken del av en anslutningen till ett nät eller inköpet av arbetsstationer är relaterat till ett viss tillämpningsområde eller ens till språkteknologiprodukter? **BOTH ARE DETECTED!**

Swetwol:
nominalfrasen: "en anslutningen" - är inkorrekt, substantivet ska bytas ut
lemma=ansluta
gen=utr
spec=indef
num=sg
case=nom

regel 23

Swetwol:
nominalfrasen: "ett viss tillämpningsområde" - är inkorrekt, adjektivet ska bytas ut
lemma=viss
gen=neu
spec=indef
num=sg
regel 20

XPOST:
nominalfrasen: "ett viss tillämpningsområde" - är inkorrekt, adjektivet ska bytas ut
lemma=viss
gen=neu
spec=ind
num=sin
regel 20

14. De allmänna principen för att konstruera en formel är vanligen sedan denna: **DETECTED!**

Swetwol:
nominalfrasen: "de allmänna principen" - är inkorrekt, determineraren ska bytas ut
lemma=de
gen=utr
spec=def
num=sg
regel 19

XPOST:
nominalfrasen: "de allmänna principen" - är inkorrekt, determineraren ska bytas ut
lemma=de
gen=utr
spec=def
num=sin
regel 19

15. Den sådan, generell Lix-tolk ser ut på följande sätt (Larsson 1987): **NOT DETECTED!**
Error analysis: *sådan* is asserted as a determiner

16. Programmet har gjort succe på de amerikanska marknaden, recension se (Rash 1991). **DETECTED!**
Swetwol:
nominalfrasen: "de amerikanska marknaden" - är inkorrekt, determineraren ska bytas ut
lemma=de
gen=utr
spec=def
num=sg
regel 19

XPOST:
nominalfrasen: "de amerikanska marknaden" - är inkorrekt, determineraren ska bytas ut
lemma=de
gen=utr
spec=def
num=sin
regel 19

17. Bakgrundskunskapen hos läsaren är det tredje aspekten i den lingvistiska kritiken av formlerna. **DETECTED!**

Swetwol:
nominalfrasen: "det tredje aspekten" - är inkorrekt, determineraren ska bytas ut
lemma=det
gen=utr
spec=def
num=sg
regel 19

18. Congletons skala är av semantisk karaktär och det krävs kvalificerade bedömare för att beräkna de metaforiska svårighetsgraden. **DETECTED!**

Swetwol:
nominalfrasen: "de metaforiska svårighetsgraden" - är inkorrekt, determineraren ska bytas ut
lemma=de
gen=utr
spec=def
num=sg
regel 19

XPOST:
nominalfrasen: "de metaforiska svårighetsgraden" - är inkorrekt, substantivet ska bytas ut
lemma=svårighetsgraden
gen=utr
spec=def
num=plu
case=nom
regel 21

19. Genomsnittligt meningslängd i antal ord. DETECTED**!**

Swetwol:
nominalfrasen: "genomsnittligt meningslängd" - är inkorrekt, adjektivet ska bytas ut
lemma=genomsnittlig
gen=utr
spec=indef
num=sg
regel 25

20. En automatiserad kontroll av denna ortografisk korrekthet är vanligtvis tekniskt enkel att genomföra. DETECTED**!**

Swetwol:

nominalfrasen: "denna ortografisk korrekthet" - är inkorrekt, determineraren ska bytas ut
lemma=denna
gen=utr
spec=indef
num=sg
regel 19

XPOST:

nominalfrasen: "denna ortografisk korrekthet" - är inkorrekt, determineraren ska bytas ut
lemma=denna
gen=utr
spec=ind
num=sin
regel 19

21. Vissa regelklasser inom denna grupp av automatiserade textgranskning är mer eller mindre entydiga, medan andra regler är mer subjektiva och fungerar bara i vissa sammanhang. **DETECTED!**

Swetwol:

nominalfrasen: "automatiserade textgranskning" - är inkorrekt, adjektivet ska bytas ut
lemma=automatisera
gen=utr
spec=indef
num=sg
regel 25

XPOST:

nominalfrasen: "automatiserade textgranskning" - är inkorrekt, adjektivet ska bytas ut
lemma=automatiserade
gen=utr
spec=ind
num=sin
regel 25

22. Dessa datorkonstruerade mening är inte avsedda som en slutlig formulering, utan är avsedda att hjälpa skribenten till nya uppslag och infallsvinklar på

sitt ämne. DETECTED**!**

Swetwol:
nominalfrasen: "dessa datorkonstruerade mening" - är inkorrekt, substantivet ska bytas ut
lemma=mening
gen=utr
spec=def
num=pl
case=nom
regel 21

XPOST:
nominalfrasen: "dessa datorkonstruerade mening" - är inkorrekt 106 determineraren och adjektivet ska bytas ut
lemma=dessa
gen=utr
spec=ind
num=sin
lemma=datorkonstruerade
gen=utr
spec=ind
num=sin
regel 22

23. Både programmen Word Perfect och Microsoft Word har i sin senare versioner för svenska inbyggda svenska synonymordböcker. **NOT DETECTED!**

Overgeneration by Swetwol:
nominalfrasen: "inbyggda svenska" - är inkorrekt, adjektivet ska bytas ut
lemma=inbygga
gen=utr
spec=indef
num=sg
regel 25

Error analysis: *sin* is pronoun which could be seen as a determiner in this case. It is not implemented, that will cause some overgeneration.

24. Genom att från början trots detta göra all programkod generell och portabel kan språkstödsfunktionerna i ett senare skede flyttas till en annat programmeringsmiljö. **NOT DETECTED!**
Error analysis: *annat* is a pronoun (in Swetwol) and the NP dt_pn_n is not implemented

XPOST:
nominalfrasen: "en annat programmeringsmiljö" - är inkorrekt, determineraren ska bytas ut
lemma=en
gen=neu
spec=ind
num=sin
regel 19

25. Hennes modell för skrivprocessen (se figur 4) är i princip detsamma som anglo-amerikanska skrivpedagogiska modellerna. **NOT DETECTED!**
Error analysis: there is a determiner missing here.

26. En konkret matrial rymmer mer eller mindre färdiga formuleringar, medan ett abstrakt endast innehåller lösryckta uttryck eller noteringar. **DETECTED!** (except *matrial*)
Swetwol:
nominalfrasen: "en konkret" - är inkorrekt, determineraren ska bytas ut
lemma=en
gen=neu
spec=indef
num=sg
regel 24

27. Sådana program gör dessutom en enklare kvantitativa kontroller av en text. **DETECTED!**

Swetwol:
nominalfrasen: "en enklare kvantitativa kontroller" - är inkorrekt , determineraren - ¨en¨ ska bytas ut
lemma=en
gen=utr
spec=indef
num=pl
regel 11

XPOST:
nominalfrasen: "en enklare kvantitativa kontroller" - är inkorrekt , determineraren - ¨en¨ ska bytas ut
lemma=en
gen=utr
spec=ind
num=plu
regel 11

28. Det viktigaste källorna till förhandsplanering av tal är följande: **DETECTED!**

Swetwol:
nominalfrasen: "det viktigaste källorna" - är inkorrekt, determineraren ska bytas ut
lemma=det
gen=utr
spec=def
num=pl
regel 19

XPOST:
nominalfrasen: "det viktigaste källorna" - är inkorrekt, determineraren ska bytas ut
lemma=det
gen=utr
spec=def
num=plu
regel 19

29. Den obefintliga tabellhanteraren från förra version är numera ersatt av en "riktig" dito. **DETECTED!**

Swetwol:
nominalfrasen: "förra version" - är inkorrekt, adjektivet ska bytas ut
lemma=förra
gen=utr
spec=indef
num=sg
regel 25

XPOST:
nominalfrasen: "förra version" - är inkorrekt, adjektivet ska bytas ut
lemma=förra
gen=utr
spec=ind
num=sin
regel 25

Overgeneration by XPOST:
nominalfrasen: "den obefintliga tabellhanteraren" - är inkorrekt, determineraren ska bytas ut
lemma=den
gen=neu
spec=def
num=plu
regel 19

30. IBM har väntat i de längsta med att introducera den nya serien i hopp om att få fram ett nytt operativsystem, nu beräknas det istället komma om sex till tolv månader. **NOT DETECTED!**
Error analysis: This NP type is not implemented

31. Det här synen, att monologen är språklig kommunikation med en själv, accepterar också Chomsky (1975). **DETECTED!** (except *synen*)

Overgeneration by Swetwol:
nominalfrasen: "det här" - är inkorrekt, determineraren ska bytas ut
lemma=det
gen=utr
spec=indef
num=sg
regel 24

Overgeneration by Swetwol:
nominalfrasen: "en själv" - är inkorrekt, determineraren ska bytas ut
lemma=en
gen=neu
spec=indef
num=sg
regel 24

32. Det första artskillnaden gäller det tysta icke-verbala tänkandet gentemot det övriga. **DETECTED!**

Swetwol:
nominalfrasen: "det första artskillnaden" - är inkorrekt, determineraren ska bytas ut
lemma=det
gen=utr
spec=def
num=sg
regel 19

33. Det har hävdats att ett graden av demokrati i ett samhälle kan avläsas genom att studera mängden av information och i hur hög grad medborgarna att använder sig av media för att uttrycka sina åsikter offentligt. **DETECTED!**

Swetwol:
nominalfrasen: "ett graden" - är inkorrekt, determineraren ska bytas ut
lemma=ett
gen=utr
spec=def
num=sg
regel 24

XPOST:
nominalfrasen: "ett graden" - är inkorrekt, determineraren ska bytas ut
lemma=ett

gen=utr
spec=def
num=sin
regel 24

34. Dessutom anser vi det vara en självklarhet i en demokratin att som medborgare äga rättigheten att granska våra politikers förehavanden. **DETECTED!**

Swetwol:
nominalfrasen: "en demokratin" - är inkorrekt, substantivet ska bytas ut
lemma=demokrati
gen=utr
spec=indef
num=sg
case=nom
regel 23

XPOST:
nominalfrasen: "en demokratin" - är inkorrekt, substantivet ska bytas ut
lemma=demokratin
gen=utr
spec=ind
num=sin
case=nom
regel 23

35. Han reste sig snabbt och var givetvis huvudperson i de våldsamma segerfest som väl knappast har slutat ännu. **DETECTED!**

Swetwol:
nominalfrasen: "de våldsamma segerfest" - är inkorrekt, substantivet ska bytas ut
lemma=segerfest
gen=utr
spec=def
num=pl
case=nom
regel 21

XPOST:
nominalfrasen: "de våldsamma segerfest" - är inkorrekt, substantivet ska bytas ut
lemma=segerfest
gen=utr

spec=def
num=plu
case=nom
regel 21

36. I STU-projektet (CIM) har man forskat kring hur man ska förbättra kommunikationen i interaktiva system m.h.a. text och bild. Speciellt lexivisuell presentationer. **NOT DETECTED!**
Error analysis: *lexivisuell* is not analysed by Swetwol.

37. En del säger att ur smältdegel mellan tv-radio och förlagsverksamhet och dataindustri kommer en ny typ av media, den totalintegrerade "New media". **NOT DETECTED!**
Error analysis: *new* is very difficult for Swetwol.

38. 1978 startade Datavision. Detta blev starten för Esselte inom den digitaliserad förlagsverksamhet. **NOT DETECTED!**
Error analysis: *digitaliserad* is not analysed by Swetwol.

Overgeneration by Swetwol:
nominalfrasen: "1978 startade datavision" - är inkorrekt 106 determineraren och adjektivet ska bytas ut
lemma=1978
gen=utr
spec=indef
num=sg
lemma=starta
gen=utr
spec=indef
num=sg
regel 22

XPOST:
nominalfrasen: "den digitaliserad förlagsverksamhet" - är inkorrekt, determineraren ska bytas ut
lemma=den
gen=utr
spec=ind
num=sin
regel 19

Overgeneration by XPOST
nominalfrasen: "startade datavision" - är inkorrekt, adjektivet ska bytas ut
lemma=startade
gen=utr
spec=ind
num=sin
regel 25

39. Enligt Erik -"Såsom andelsägare av den Svenska nationalförmögenhet är varje svensk medborgare mångmiljonär. **DETECTED!**

Overgeneration by Swetwol:
nominalfrasen: "enligt erik" - är inkorrekt, adjektivet ska bytas ut

lemma=enlig
gen=?
spec=?
num=sg
regel 25

XPOST:
nominalfrasen: "den svenska nationalförmögenhet" - är inkorrekt, substantivet ska bytas ut
lemma=nationalförmögenhet
gen=utr
spec=def
num=sin
case=nom
regel 21

40. När jag frågar vad IBM gör inom multimediaområdet ser Sten Kallin lite frågande ut och undrar vad det är som är så nytt med detta multimedia. **DETECTED!**

Swetwol:
nominalfrasen: "detta multimedia" - är inkorrekt, substantivet ska bytas ut
lemma=multimedia
gen=neu
spec=def
num=sg
case=nom
regel 23

41. Vad man menar med det är att många hypersystem utger sig för lagra information som en hyperdokument, t ex om alla böcker i ett bibliotek. **DETECTED!**

Swetwol:
nominalfrasen: "en hyperdokument" - är inkorrekt, determineraren ska bytas ut
lemma=en
gen=neu
spec=indef
num=_98418
regel 24

42. Sedan är det beställarens sak att köpa ett EyesCreamsystem för att sedan själv göra det slutgiltiga lösningen. **DETECTED!**

Swetwol:
nominalfrasen: "det slutgiltiga lösningen" - är inkorrekt, determineraren ska bytas ut
lemma=det
gen=utr
spec=def
num=sg
regel 19

Overgeneration by Swetwol:
nominalfrasen: "det beställarens" - är inkorrekt, determineraren ska bytas ut
lemma=det

gen=utr
spec=def
num=sg
regel 24

XPOST:
nominalfrasen: "det slutgiltiga lösningen" - är inkorrekt, determineraren ska bytas ut
lemma=det
gen=utr
spec=def
num=sin
regel 19

Overgeneration by XPOST:
nominalfrasen: "det beställarens" - är inkorrekt, determineraren ska bytas ut
lemma=det
gen=utr
spec=ind
num=sin
regel 24

43. Negroponte, som är chef för MIT's medialabb, har uttryckt visionen att de traditionella medierna skall smälta ihop i ett nytt media. **DETECTED!**

Swetwol:
nominalfrasen: "ett nytt media" - är inkorrekt, substantivet ska bytas ut
lemma=media
gen=neu
spec=indef
num=sg
case=nom
regel 21

Overgeneration by Swetwol:
nominalfrasen: "uttryckt visionen" - är inkorrekt, adjektivet ska bytas ut
lemma=uttrycka
gen=utr
spec=def
num=sg
regel 25

XPOST:
nominalfrasen: "ett nytt media" - är inkorrekt, substantivet ska bytas ut
lemma=media
gen=neu
spec=ind
num=sin
case=nom
regel 21

44. Samma statiska text och grafikinformation för prepresentation på papper måste, med ev. tillägg av dynamisk ljud och video kunna presenteras elektroniskt. **DETECTED!**

Swetwol:
nominalfrasen: "dynamisk ljud" - är inkorrekt, adjektivet ska bytas ut
lemma=dynamisk
gen=neu
spec=indef
num=_59964
regel 25

Overgeneration by Swetwol:
nominalfrasen: "samma statiska text" - är inkorrekt, adjektivet ska bytas ut
lemma=statisk
gen=utr
spec=indef
num=sg
regel 20

XPOST:
nominalfrasen: "dynamisk ljud" - är inkorrekt, adjektivet ska bytas ut
lemma=dynamisk
gen=neu
spec=ind
num=plu
regel 25

Overgeneration by XPOST:
nominalfrasen: "samma statiska text" - är inkorrekt, adjektivet ska bytas ut
lemma=statiska
gen=utr
spec=ind
num=sin
regel 20

45. Dessa studier skall ske genom tvärvetenskaplig samverkan vid utformning av prototyper till nästa generations gränssnitt för interaktiva TV. **DETECTED!**

Swetwol:
nominalfrasen: "interaktiva tv" - är inkorrekt, adjektivet ska bytas ut

lemma=interaktiv
gen=utr
spec=indef
num=sg
regel 25

XPOST:
nominalfrasen: "interaktiva tv" - är inkorrekt, adjektivet ska bytas ut
lemma=interaktiva
gen=utr
spec=ind
num=sin
regel 25

46. Den tidigare forskning inom Skandinavien har främst varit inriktad på att ytveckla olika metoder för s.k. deltagande design. **DETECTED!**

Swetwol:
nominalfrasen: "den tidigare forskning" - är inkorrekt, determineraren ska bytas ut
lemma=den
gen=utr
spec=indef
num=sg
regel 19

XPOST:
nominalfrasen: "den tidigare forskning" - är inkorrekt, determineraren ska bytas ut
lemma=den
gen=utr
spec=ind
num=sin
regel 19